

Exhibit 4

U.S. Patent No. 7,370,011

Charles Schwab Corporation



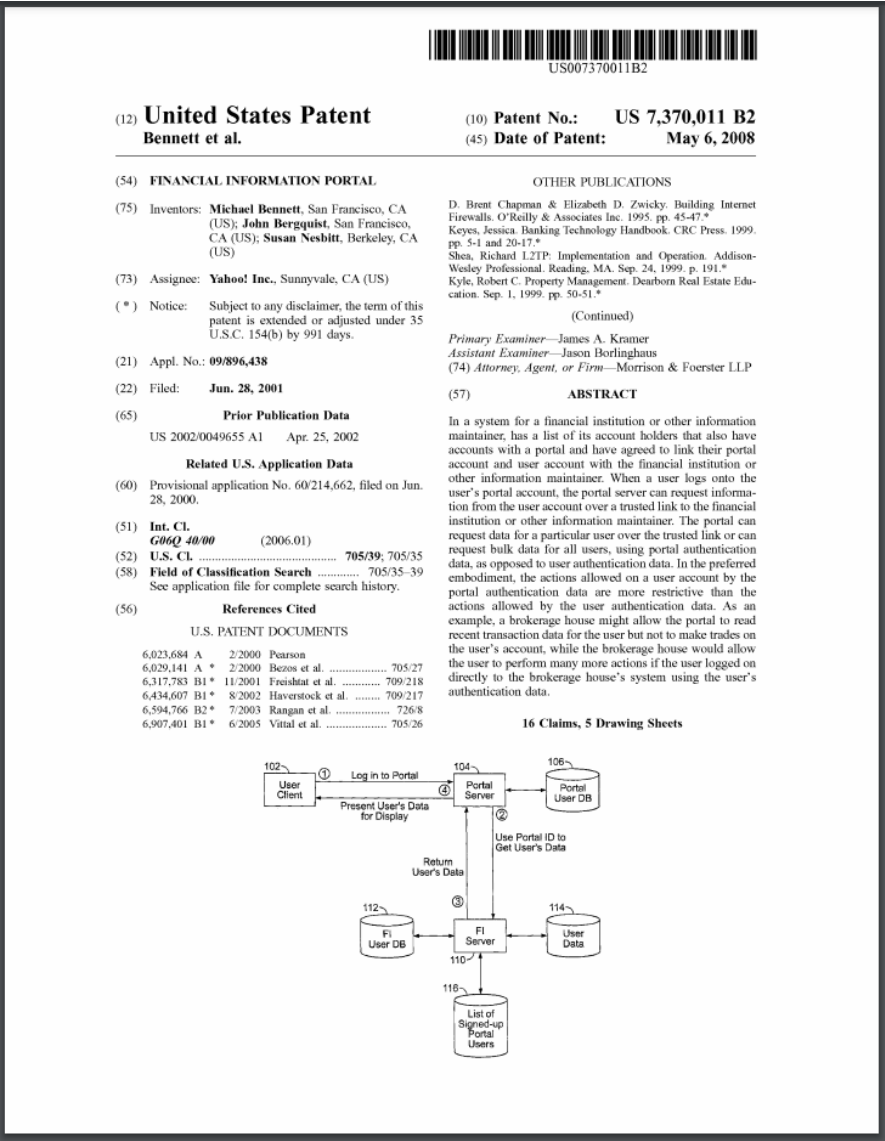
nelson
bumgardner
albritton

Claim Chart for Representative Claim 7

OVERVIEW



nelson
bumgardner
albritton



Title: FINANCIAL INFORMATION PORTAL

Priority Date: June 28, 2000

Filing Date: June 28, 2001

Issue Date: May 6, 2008

Expiration Date: March 15, 2024

(57) **ABSTRACT**

In a system for a financial institution or other information maintainer, has a list of its account holders that also have accounts with a portal and have agreed to link their portal account and user account with the financial institution or other information maintainer. When a user logs onto the user's portal account, the portal server can request information from the user account over a trusted link to the financial institution or other information maintainer. The portal can request data for a particular user over the trusted link or can request bulk data for all users, using portal authentication data, as opposed to user authentication data. In the preferred embodiment, the actions allowed on a user account by the portal authentication data are more restrictive than the actions allowed by the user authentication data. As an example, a brokerage house might allow the portal to read recent transaction data for the user but not to make trades on the user's account, while the brokerage house would allow the user to perform many more actions if the user logged on directly to the brokerage house's system using the user's authentication data.

Representative Claim 7

A computer readable medium storing instructions for execution in a computer, the medium when executed by a computer performing the method comprising:

accepting a connection at an institution server, the connection initiated by a user following a link from a portal, the link including a user identification;

responsive to the connection, enabling the user to authenticate with the institution server using user-institution authentication data;

responding to the authentication by associating the user identification with the portal; and

servicing a request by the portal, after authenticating the portal using portal authentication data, by providing, to the portal, data of the user at the institution, wherein the user-institution authentication data and the portal authentication data are not the same data.

CLAIM CHART



nelson
bumgardner
albritton

References Cited

Upon information and belief, Charles Schwab Corporation (“Charles Schwab”) has and continues to make, use, sell, offer for sale, and/or import the Charles Schwab interface allowing users to associate an account with a third-party service (such as Plaid) (hereinafter, the “Schwab interface”), which is software that is capable of practicing the steps described in the claims of U.S. Patent No. 7,370,011 (the “’011 Patent”).

The following chart presents R2 Solutions’ analysis of the Schwab interface based on publicly available information.

The citations in the chart refer to the following publicly available documents, which are incorporated by reference as if fully set forth herein:

- [1] *What is Plaid?*, <https://plaid.com/what-is-plaid/>.
- [2] *Link Overview*, <https://plaid.com/docs/link/#introduction-to-link>.
- [3] *Plaid Link Demo*, <https://plaid.com/demo/?countryCode=US&language=en&product=transactions>.
- [4] *How We Handle Data*, <https://plaid.com/how-we-handle-data/>.
- [5] *How It Works*, <https://plaid.com/docs/quickstart/#how-it-works>.

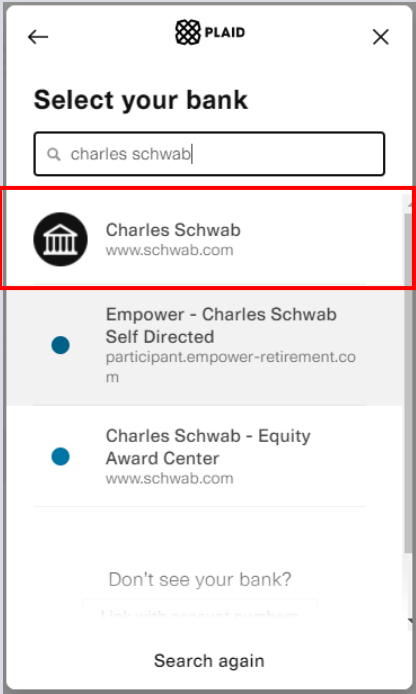
References Cited

- [6] *Demystifying Screenless Exchange*, <https://fin.plaid.com/articles/demystifying-screenless-exchange/>.
- [7] Lisa Schidler, *Tired of Having Their Screens Scraped, Schwab and Fidelity Launch API Initiatives to Curtail the Practice—Rewarding Some, but Not All Scrapers with Cleaner Data*, RIABiz (May 28, 2020, 11:45 PM), <https://riabiz.com/a/2020/5/29/tired-of-having-their-screens-scraped-schwab-and-fidelity-launch-api-initiatives-to-curtail-the-practice-rewarding-some-but-not-all-scrapers-with-cleaner-data>.
- [8] *Product Endpoints: Identity*, <https://plaid.com/docs/api/products/#identity>.

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
<p>A computer readable medium storing instructions for execution in a computer, the medium when executed by a computer performing the method comprising:</p>	<p>Charles Schwab allows customers to connect their bank accounts with Plaid. On information and belief, this involves Charles Schwab taking steps on its servers to associate the user's bank account with the user's Plaid account. Plaid is a software portal that allows users to connect applications to their bank accounts. <i>See, e.g.:</i></p> <div data-bbox="784 464 1490 682"> <p>What is Plaid?</p> <p>Plaid makes it easy to securely connect your bank to the apps you want to use</p> </div> <div data-bbox="784 696 1490 1232"> <p>Connecting your bank to your apps</p> <p>We power thousands of the apps that people rely on to manage their financial lives.</p> <ul style="list-style-type: none"> • Venmo (peer-to-peer payments) • Betterment (automated investing) • Chime (online banking) • Dave (earned wage access) • And thousands more... </div> <div data-bbox="1803 468 2440 1225"> <p>How it works</p> <p>We connect to 11,000 financial institutions across the United States, Canada, and Europe. With Plaid, connecting your bank account is easy:</p> <ul style="list-style-type: none"> • Step 1 When you sign up with a Plaid-powered app, you'll be able to select your financial institution from a list. Then, enter your login and password. • Step 2 In a matter of seconds, we encrypt the data you've chosen to share (for instance, your account balance) and securely share it with the app you want to use. We never share your login and password with the app. • Step 3 We work behind the scenes to build a secure, ongoing connection between the app and your bank. </div> <p><i>What is Plaid?</i></p>

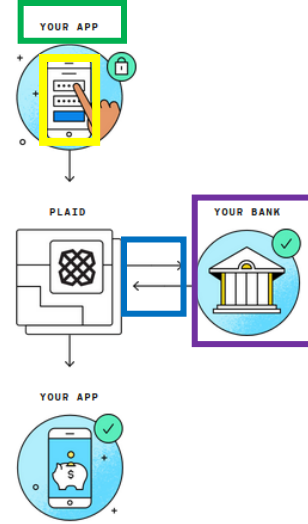
Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
<p>A computer readable medium storing instructions for execution in a computer, the medium when executed by a computer performing the method comprising:</p>	<p>Charles Schwab allows customers to connect their bank accounts with Plaid. On information and belief, this involves Charles Schwab taking steps on its servers to associate the user’s bank account with the user’s Plaid account. Plaid is a software portal that allows users to connect applications to their bank accounts. <i>See, e.g.:</i></p> <div data-bbox="741 486 1154 1173">  </div> <div data-bbox="1363 708 2446 1028"> <p>Plaid Link is the client-side component that your users will interact with in order to link their accounts to Plaid and allow you to access their accounts via the Plaid API.</p> <p>Plaid Link will handle credential validation, multi-factor authentication, and error handling for each institution that we support. Link works across all modern browsers and platforms, including <u>web</u>, <u>iOS</u>, <u>Android</u>, and <u>mobile webviews</u>. For a full list of supported platforms, see <u>Link client libraries</u>.</p> </div> <p><i>Link Overview</i></p> <p><i>Plaid Link Demo</i></p>

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
<p>A computer readable medium storing instructions for execution in a computer, the medium when executed by a computer performing the method comprising:</p>	<p>On information and belief, Charles Schwab uses an API that allows Plaid to access customer data using a token-based authentication process. See, e.g.:</p> <p><u>Tired of having their screens scraped, Schwab and Fidelity launch API initiatives to curtail the practice -- rewarding some, but not all scrapers with cleaner data</u></p> <p>"APIs are a best practice in the industry and take a token-based approach, which enables clients to authorize third parties to download requested account information on their behalf in an encrypted form, without storing their usernames and passwords."</p> <p>Shidler</p>



Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
<p>accepting a connection at an institution server, the connection initiated by a user following a link from a portal, the link including a user identification;</p>	<p>A user may access Plaid (an example of a “portal”) through a financial application. The user may then initiate a connection with Charles Schwab (i.e., an “institution”) by following a link displayed on the Plaid interface. See, e.g.:</p> <div data-bbox="1136 415 1936 1222"> <p>How we handle your data</p> <p>With Plaid, connecting your financial accounts to an app is simple. Here's what happens on our end.</p> <ol style="list-style-type: none"> 1 To link your financial accounts to an app like Venmo, simply provide the username and password associated with those financial accounts. 2 Once we've verified your ownership of your accounts, we retrieve your account information from your financial institution. 3 We then securely process and share your information with the app you're using and establish a secure connection that you control.  <p>The diagram illustrates the process of connecting a financial app to a bank via Plaid. It shows three main components: 'YOUR APP' (represented by a smartphone icon), 'PLAID' (represented by a server icon), and 'YOUR BANK' (represented by a bank building icon). The process follows three steps: 1. The user provides credentials to the app. 2. Plaid verifies the user's ownership and retrieves account information from the bank. 3. The information is securely processed and shared back with the app, establishing a secure connection.</p> </div> <p><i>How We Handle Data</i></p>

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
accepting a connection at an institution server, the connection initiated by a user following a link from a portal, the link including a user identification;	<p>A user may access Plaid (an example of a “portal”) through a financial application. The user may then initiate a connection with Charles Schwab (i.e., an “institution”) by following a link displayed on the Plaid interface. See, e.g.:</p> <div><p>Screenless exchange</p><div><div><div>1. Credentials sent from user device to bank via Plaid. Bank tokenizes, stores, and returns a credentials token without verifying whether credentials are valid.</div><div>2. User device sends credentials token to Plaid.</div><div>3. Plaid sends credentials token to bank. If verification is successful, bank stores an access token that is also sent to Plaid.</div><div>4. User is prompted to accept Terms and informed of data the application intends to access.</div><div>5. Plaid stores institution access key and credential token in a secure database.</div><div>6. Application developer receives API access key.</div><div>7. Developer uses API access key to request data from Plaid, including any subsequent requests.</div><div>8. Plaid uses the institution access token associated with the API access key to request data from bank.</div><div>9. Plaid returns data to application developer, which then displays data on user’s device.</div></div><p>The diagram illustrates the 'Screenless exchange' process. It features five main components: Application developer (top left), User device (top right), Plaid (center), Financial Institution (bottom right), and Secure database (bottom center). The process is numbered 1 through 9. 1. A green box highlights the step: '1. Credentials sent from user device to bank via Plaid. Bank tokenizes, stores, and returns a credentials token without verifying whether credentials are valid.' 2. An arrow points from the User device to Plaid. 3. An arrow points from Plaid to the Financial Institution. 4. An arrow points from the Financial Institution back to the User device. 5. An arrow points from Plaid to the Secure database. 6. An arrow points from the Secure database to the Application developer. 7. An arrow points from the Application developer to Plaid. 8. An arrow points from Plaid to the Financial Institution. 9. An arrow points from Plaid to the Application developer.</p></div><p>In digital financial services, Plaid presents one leading solution. In this flow, <u>an app connects with a bank via a trusted third-party intermediary (in this case, Plaid).</u> <u>The user inputs his or her credentials, which are passed directly to the bank via the trusted intermediary.</u> The application never sees or stores user credentials, significantly reducing the potential for credential compromise and fraud.</p><p><i>Demystifying Screenless Exchange</i></p></div>

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
accepting a connection at an institution server, the connection initiated by a user following a link from a portal, the link including a user identification;	<p>For example, as the Plaid demo shows, a user can input the username and password for their Charles Schwab account. This information is then sent to Charles Schwab when the user selects the submit button. In other words, the user initiates a connection at the institution server by following a link from the portal, and the link includes user identification. See, e.g.:</p> <div><div><p>← PLAID ×</p><div> Charles Schwab www.schwab.com</div><p>Enter your credentials</p><div><input type="text" value="Login ID"/> user_good</div><div><input type="password" value="Password"/></div><div>Submit</div><div>Reset password</div></div><p>→</p><div><p>PLAID ×</p><div></div><p>Success!</p><p>Your account has been successfully linked to Plaid Demo</p><div>Continue</div></div></div> <p>Plaid Link Demo</p>

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
<p>responsive to the connection, enabling the user to authenticate with the institution server using user-institution authentication data;</p>	<p>After Plaid connects to the Charles Schwab server, the user may be authenticated with the Charles Schwab server using user-institution authentication data (e.g., the username/password input into the Plaid interface). As the flow diagram below shows, the user’s bank credentials are sent to the bank servers and then the user’s account is verified. See, e.g.:</p> <div data-bbox="988 468 2135 1245"> <p>Screenless exchange</p> <p>1. Credentials sent from user device to bank via Plaid. Bank tokenizes, stores, and returns a credentials token without verifying whether credentials are valid.</p> <p>2. User device sends credentials token to Plaid.</p> <p>3. Plaid sends credentials token to bank. If verification is successful, bank stores an access token that is also sent to Plaid.</p> <p>4. User is prompted to accept Terms and informed of data the application intends to access.</p> <p>5. Plaid stores institution access key and credential token in a secure database.</p> <p>6. Application developer receives API access key.</p> <p>7. Developer uses API access key to request data from Plaid, including any subsequent requests.</p> <p>8. Plaid uses the institution access token associated with the API access key to request data from bank.</p> <p>9. Plaid returns data to application developer, which then displays data on user's device.</p> </div>

Demystifying Screenless Exchange

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
responding to the authentication by associating the user identification with the portal; and	<p>In response to authentication of the user’s bank credentials, the Charles Schwab servers associate the user identification with the portal. For example, the bank server saves an access token that is shared with the portal. The access token is linked with the user’s account, which includes a user identification. See, e.g.:</p> <div><p>Screenless exchange</p><p>The diagram illustrates a screenless exchange process with five main components: Application developer (top left), User device (top right), Plaid (center), Financial institution (bottom right), and Secure database (bottom left). The process is numbered 1 through 9:</p><ul style="list-style-type: none">1. User device sends credentials to Financial institution.2. User device sends credentials token to Plaid.3. Plaid sends credentials token to Financial institution.4. User is prompted to accept Terms and informed of data the application intends to access.5. Plaid stores institution access key and credential token in a secure database.6. Application developer receives API access key.7. Developer uses API access key to request data from Plaid, including any subsequent requests.8. Plaid uses the institution access token associated with the API access key to request data from bank.9. Plaid returns data to application developer, which then displays data on user's device.</div>

Demystifying Screenless Exchange

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
responding to the authentication by associating the user identification with the portal; and	<p>The association between the user identification and the access token (and thereby the portal) is confirmed based on the response that the portal receives when it uses the access token to retrieve user identity information. See, e.g.:</p> <div><p>/identity/get Retrieve identity data</p><p>The <u>/identity/get</u> endpoint allows you to retrieve various account holder information on file with the financial institution, including names, emails, phone numbers, and addresses. Only name data is guaranteed to be returned; other fields will be empty arrays if not provided by the institution.</p></div> <div><pre>1 // Pull Identity data for an Item 2 Response<IdentityGetResponse> response = client().se 3 new IdentityGetRequest("ACCESS_TOKEN") 4).execute(); 5 List<IdentityGetResponse.AccountWithOwners> accounts 6 for (IdentityGetResponse.AccountWithOwners account : 7 List<IdentityGetResponse.Identity> identities = ac 8 }</pre></div> <p><i>Product Endpoints: Identity</i></p>

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
responding to the authentication by associating the user identification with the portal; and	<div><div><div>Response fields and example</div><div>Collapse all ^</div><div><div><div>accounts</div><div>[object]</div><div>The accounts for which Identity data has been requested</div><div>Hide object</div></div><div><div>account_id</div><div>string</div><div>Plaid's unique identifier for the account. This value will not change unless Plaid can't reconcile the account with the data returned by the financial institution. This may occur, for example, when the name of the account changes. If this happens a new account_id will be assigned to the account.</div><div>The account_id can also change if the access_token is deleted and the same credentials that were used to generate that access_token are used to generate a new access_token on a later date. In that case, the new account_id will be different from the old account_id.</div><div>Like all Plaid identifiers, the account_id is case sensitive.</div></div></div></div><div><div>API Object</div><div><pre>1 { 2 "accounts": [3 { 4 "account_id": "BxBXxLj1m4HMXBm9WZZmCWVbPjX16EHwv99vp 5 "balances": { 6 "available": 100, 7 "current": 110, 8 "iso_currency_code": "USD", 9 "limit": null, 10 "unofficial_currency_code": null 11 }, 12 "mask": "0000", 13 "name": "Plaid Checking", 14 "official_name": "Plaid Gold Standard 0% Interest Ch 15 "owners": [16</pre></div></div></div> <div>Product Endpoints: Identity</div>

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
<p>servicing a request by the portal, after authenticating the portal using portal authentication data, by providing, to the portal, data of the user at the institution, wherein the user-institution authentication data and the portal authentication data are not the same data.</p>	<p>Charles Schwab servers are configured to service a request by the portal, after authenticating the portal using portal authentication data. For example, on information and belief, Charles Schwab's API is configured to enable tokenized data exchange with Plaid. A user is authenticated with a token (i.e., portal authentication data) rather than the user's institution username/password (i.e., user-institution authentication data) when the portal requests user data from Charles Schwab. See, e.g.:</p> <p><u>Tired of having their screens scraped, Schwab and Fidelity launch API initiatives to curtail the practice -- rewarding some, but not all scrapers with cleaner data</u></p> <p>"APIs are a best practice in the industry and take a <u>token-based</u> approach, which enables clients to authorize third parties to <u>download requested account information</u> on their behalf in an encrypted form, without storing their <u>usernames and passwords</u>."</p> <p>Shidler</p>

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
<p>servicing a request by the portal, after authenticating the portal using portal authentication data, by providing, to the portal, data of the user at the institution, wherein the user-institution authentication data and the portal authentication data are not the same data.</p>	<p>Charles Schwab servers are configured to service a request by the portal, after authenticating the portal using portal authentication data. For example, on information and belief, Charles Schwab’s API is configured to enable tokenized data exchange with Plaid. A user is authenticated with a token (i.e., portal authentication data) rather than the user’s institution username/password (i.e., user-institution authentication data) when the portal requests user data from the bank. See, e.g.:</p> <div data-bbox="988 515 2061 1243"> <h3>Screenless exchange</h3> <ol style="list-style-type: none"> 1. Credentials sent from user device to bank via Plaid. Bank tokenizes, stores, and returns a credentials token without verifying whether credentials are valid. 2. User device sends credentials token to Plaid. 3. Plaid sends credentials token to bank. If verification is successful, bank stores an access token that is also sent to Plaid. 4. User is prompted to accept Terms and informed of data the application intends to access. 5. Plaid stores institution access key and credential token in a secure database. 6. Application developer receives API access key. 7. Developer uses API access key to request data from Plaid, including any subsequent requests. 8. Plaid uses the institution access token associated with the API access key to request data from bank. 9. Plaid returns data to application developer, which then displays data on user’s device. </div>

Demystifying Screenless Exchange

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
<p>servicing a request by the portal, after authenticating the portal using portal authentication data, by providing, to the portal, data of the user at the institution, wherein the user-institution authentication data and the portal authentication data are not the same data.</p>	<p>The Plaid API illustrates how requests are made by the portal and how the financial institution (e.g., Charles Schwab) returns data of the user in response to authentication using portal authentication data that is different from the user-institution data. As mentioned in previous slides, the user's institution credentials (i.e., user-institution authentication data) are used by plaid to obtain an access token (i.e., portal authentication data). See, e.g.:</p> <p>Once you have a <code>link_token</code>, you can use it to initialize <code>Link</code>. <code>Link</code> is a drop-in client-side module available for web, iOS, and Android that handles the authentication process. The Quickstart uses <code>Link</code> on the web, which is a pure JavaScript integration that you trigger via your own client-side code. <u>This is what your users use to log into their financial institution accounts.</u></p> <p>After a user submits <u>their credentials</u> within <code>Link</code>, <code>Link</code> provides you with a <code>public_token</code> via the <code>onSuccess</code> callback. The code below shows how the Quickstart passes the <code>public_token</code> from client-side code to the server. Both React and vanilla JavaScript examples are shown.</p> <p>Next, on the server side, the Quickstart calls <u><code>/item/public_token/exchange</code></u> to obtain an <code>access_token</code>, as illustrated in the code excerpt below. <u>The <code>access_token</code></u> uniquely identifies an <code>Item</code> and is a required argument for most Plaid API endpoints. In your own code, you'll need to securely store your <code>access_token</code> in order to make API requests for that <code>Item</code>.</p> <p><i>How It Works</i></p>

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
<p>servicing a request by the portal, after authenticating the portal using portal authentication data, by providing, to the portal, data of the user at the institution, wherein the user-institution authentication data and the portal authentication data are not the same data.</p>	<p>Plaid may then use the access token to make an API call to Charles Schwab's servers and Charles Schwab will return data of the user. In other words, after the portal authenticates itself by submitting the access token (i.e., portal authentication data) to Charles Schwab, Charles Schwab provides account data of the user. See, e.g.:</p> <div data-bbox="1174 455 1964 1235"> <p><u>Making API requests</u></p> <p>Now that we've gone over the Link flow and token exchange process, we can explore what happens when you press a button in the Quickstart to make an API call. As an example, we'll look at the Quickstart's call to <code>/accounts/get</code>, which retrieves basic information, such as name and balance, about the accounts associated with an Item. The call is fairly straightforward and uses the <code>access_token</code> as a single argument to the Plaid client object.</p> <pre> /accounts/get Ruby 1 # Retrieve an Item's accounts 2 get '/accounts' do 3 begin 4 product_response = client.accounts.get(access_token) 5 pretty_print_response(product_response) 6 content_type :json 7 { accounts: product_response }.to_json 8 rescue Plaid::PlaidAPIError => e 9 error_response = format_error(e) 10 pretty_print_response(error_response) 11 content_type :json 12 error_response.to_json 13 end 14 end </pre> <p><i>How It Works</i></p> </div>

Claim Chart – Claim 7

Claim Feature	Evidence from Charles Schwab
<p>servicing a request by the portal, after authenticating the portal using portal authentication data, by providing, to the portal, data of the user at the institution, wherein the user-institution authentication data and the portal authentication data are not the same data.</p>	<p>Plaid may then use the access token to make an API call to Charles Schwab's servers and Charles Schwab will return data of the user. In other words, after the portal authenticates itself by submitting the access token (i.e., portal authentication data) to Charles Schwab, Charles Schwab provides account data of the user. See, e.g.:</p> <div data-bbox="1065 468 2051 1245"> <p>Example response data:</p> <pre> /accounts/get response 1 { 2 "accounts": [3 { 4 "account_id": "A3wenK5EQRfKlnx1BbVXtPw9gyazDWu1EdaZD", 5 "balances": { 6 "available": 100, 7 "current": 110, 8 "iso_currency_code": "USD", 9 "limit": null, 10 "unofficial_currency_code": null 11 }, 12 "mask": "0000", 13 "name": "Plaid Checking", 14 "official_name": "Plaid Gold Standard 0% Interest Checking", 15 "subtype": "checking", 16 "type": "depository" </pre> </div> <p><i>How It Works</i></p>

END



nelson
bunyardner
albritton